

巧学巧用

Borland C++ 5.0 for Windows 95

钱子明 应 晓 主编

最新出版



電子工業出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

巧学巧用

Borland C++ 5.0 for Windows 95

钱子明 应 晓 主编

电子工业出版社

内 容 简 介

本书是为利用 Borland C++ 4.5 和 5.0 进行 C 和 C++ Windows 95 程序设计的人员编写的。首先介绍 Windows 95 的新特点,介绍 Borland C++ 4.5 和 5.0 集成环境 (IDE)、工程管理器 and 调试器的用法;然后介绍如何用 C 编写 16 位的 Windows 95 应用程序和用 C 编写 32 位的 Windows95 应用程序的技术,介绍如何用 C++ 的 ObjectWindows 编写 Window 应用程序;最后说明用 OWL 设计 Windows 95 OLE 2 应用程序的技术。本书内容详实,示例说明性好,讲述由浅入深,适合于广大软件开发人员、软件用户及大专院校师生、计算机爱好者使用。

巧学巧用 Borland C++ 5.0 for Windows 95

殷子明 应 晓 主编

责任编辑:冯文全(特约) 姜宏斌

电子工业出版社

北京市海澱区万寿路 132 信箱 (100036)

电子工业出版社发行 各地新华书店经销

顺义县天竺颖华印刷厂印刷

*

开本: 787 × 1092 毫米 1/16 印张: 34 字数: 868 千字

1996 年 3 月第一版 1996 年 3 月北京第一次印刷

印数: 8000 册 定价: 46.00 元

ISBN 7-5053-3336-4/TP · 1271

出版说明

近几年来,随着计算机的发展及其广泛普及,它在人们的工作和生活中发挥着越来越大的作用。作为计算机“灵魂”的软件的发展也是日新月异,一些功能强大、界面友好、易学易用的优秀软件层出不穷,同时一些深受人们欢迎的流行软件的版本也不断更新、提高。

随着这些优秀软件的推出和更新,一些软件开发人员、软件用户及广大的计算机爱好者,都需要了解和学习这其中的新概念、新技术,需要尽快地熟悉并精通这些软件的操作使用 and 开发的方法与经验。为满足广大读者的迫切需求,我社特邀请了具有丰富教学经验的计算机专家,精心策划、编写了这套最新流行实用软件“巧学巧用”丛书。

这套丛书的主要特点是从最基本的概念入手,以简单易懂的解说,辅以大量的典型应用程序实例及微机运行屏幕显示图例,由浅入深地详细而系统地介绍各种最新流行实用软件的功能、特点、使用和编程方法与技巧。这套书内容翔实,循序渐进,图文并茂,给出了大量的学习经验和使用、编程技巧。因此,读者可以花最少的学习时间,以最佳的捷径,掌握并精通这些流行应用软件;这套书既可作为从零开始的读者的入门向导,又包含了技术内容的深入剖析,把程序设计的专家经验与软件复杂的技术细节紧密地融合在一起,因此,对有经验的程序员来说,也很有帮助。书中的大量例程都是软件专业人员精心设计开发的,并经过上机调试、验证。这套丛书中的大部分都有配套练习、演示磁盘,收录了书中大量的例程和相关的开发工具软件,读者可到新华书店购买或与电子工业出版社发行部软件科联系邮购。

首批计划推出的有以下九种:

巧学巧用 Visual C++ 2.X for Windows 95

巧学巧用 Visual C++ MFC for Windows 95

巧学巧用 Borland C++ 5.0 for Windows 95

巧学巧用 Borland C++ OWL for Windows 95

巧学巧用 Visual FoxPro 3.0 for Windows 95

巧学巧用 Visual Basic 4.0 for Windows 95

巧学巧用中文 Windows 95

巧学巧用 Delphi for Windows

巧学巧用中文 dBASE 5 for Windows

今后我们还将不断推出其它的新的流行实用软件方面的图书,欢迎广大读者和我们联系,提出宝贵意见和建议,联系地址:北京万寿路 173 信箱(100036),联系人:文宏武。

电子工业出版社

1995 年 9 月

173 130/06

前 言

Borland C++ 5.0 是 Borland 公司 C++ 产品的一个重要版本,从该版本开始,直接支持 Windows 95 操作系统应用程序的开发,能很好地利用 Windows 95 的新特色,Borland C++ 5.0 在 Borland C++ 4.0 的基础上,新增加和增强了如下功能:

- 能开发 Windows 95 应用程序,支持 Windows 95 的新性能。
- Borland C++ 5.0 支持用 ObjectWindows 和 ObjectComponent Framework 的 OLE 2。这些类帮助 Windows 程序员简化集成 Windows 应用程序和 OLE。
- 为了使 Windows 应用程序与 OLE 接口,Borland 还创建了名为 BOLE.DLL 的动态链接库。该动态链接库使程序员简化与现有的 Windows OLE 2 结构接口。
- 有了 Borland C++ 5.0 和 OWL,用户的应用程序可以支持如下的 OLE 功能:
 - 自动化控制器
 - 自动化服务器
 - 复合文件存储
 - In Place 编辑
 - 可链接和可嵌入的包容器
 - OLE 剪贴板支持
 - OLE 拖放支持
 - OLE 用户界面支持
 - 注册
 - 类型库
 - 本地化字符串的使用

本书是为利用 Borland C++ 5.0 进行 C 和 C++ Windows 95 程序设计的人员编写的。首先介绍 Windows 95 的新特点,介绍 Borland C++ 5.0 集成环境(IDE)、工程管理器 and 调试器的用法;然后介绍用 C 编写 16 位的 Windows 95 应用程序和用 C 编写 32 位的 Windows 95 应用程序的技术,介绍 C++ 的 ObjectWindows 编写 Window 应用程序;最后说明用 OWL 设计 Windows 95 OLE 2 应用程序的技术。本书内容详实,示例说明性好,讲述由浅入深,适合于广大计算机软件开发人员、软件用户及大专院校师生、计算机爱好者使用。

参加本书编写工作的有:吕建忠、刘和力、马明、周畅、王冬和王冰等。程红梅、胡宏芳和章明辉同志等十多人负责了本书程序的编写和调试工作。郑兴忠和周燕同志负责了本书图的制作和处理。在此表示感谢。

目 录

第一章 Windows 95 概览	1
1.1 Windows 95 的最佳使用对象	1
1.2 主要新特性	1
1.2.1 易用性	2
1.2.2 速度和功能	2
1.2.3 兼容性	3
1.3 Windows 95 的用户接口	3
1.3.1 工作桌面	4
1.3.2 My Computer	4
1.3.3 Network Neighborhood(网络邻居)	5
1.3.4 Recycle Bin(回收站)	6
1.3.5 Wizard(向导)	6
1.3.6 一个新的帮助系统	8
1.3.7 Windows Explorer	8
1.3.8 统一的控制中心	9
1.3.9 寻找文件夹或文件	10
1.3.10 统一的打印机控制	10
1.3.11 功能更强的字体管理和预览	11
1.3.12 长文件名	11
1.3.13 兼容性	12
1.4 Windows 95 的基层系统体系	12
1.5 其它方面的改进	13
第二章 Borland C++ 集成开发环境	14
2.1 Borland C++ IDE	14
2.2 快捷指导	15
2.2.1 启动	15
2.2.2 Setting Notebooks(设置记事本)	20
2.2.3 加速菜单	21
2.2.4 加速条(Speed Bar)	22
2.2.5 联机帮助	24
2.2.6 文本编辑器	25
2.2.7 消息窗口	26
2.2.8 工具菜单	27
2.2.9 集成 GUI 调试器	28

2.3 编译本书程序	31
2.3.1 发现错误怎么办	31
2.3.2 在 Windows 下进行编译	32
2.3.3 在 DOS 下进行编译	32
2.3.4 编译 IDE 工程文件	33
2.4 总 结	33
第三章 IDE 工程管理器	34
3.1 什么是工程管理器	34
3.2 建立一个工程	35
3.2.1 建立一个多任务工程	37
3.2.2 转换旧的工程	38
3.2.3 将工程转化为制作文件(makefiles)	38
3.2.4 改变工程视图	38
3.3 建立一个工程与建立工程的一部分	40
3.4 编辑工程树	42
3.4.1 使用 TargetExpert 编辑任务属性	43
3.4.2 编辑节点属性	43
3.4.3 增加和删除一个节点	44
3.4.4 增加和删除任务	45
3.4.5 移动节点和任务	47
3.4.6 拷贝节点	48
3.5 使用 Source Pool	48
3.6 设置工程选项	49
3.6.1 Local Override	50
3.6.2 使用 Style Sheet	51
3.6.3 将一个 Style Sheet 与一个节点相连	52
3.6.4 生成一个 Style Sheet	53
3.6.5 编辑 Style Sheet	53
3.6.6 共享 Style Sheet	57
3.6.7 查看工程中的选项	58
3.7 转换器	59
3.7.1 安装一个转换器	60
3.7.2 使用 SpeedMenu 中的 Special 命令	62
3.7.3 安装观察器和工具	63
第四章 集成调试器	65
4.1 错误类型	65
4.1.1 编译时间错误	65
4.1.2 运行时间错误	66
4.1.3 逻辑错误	66

4.2 生成调试信息	67
4.3 指明程序参数	67
4.4 控制程序执行	68
4.4.1 监视程序的输出	68
4.4.2 单步执行程序(Step over code)	69
4.4.3 跟踪执行代码	70
4.4.4 单步跳过一段程序代码	70
4.4.5 停止程序运行	71
4.4.6 重新开始	72
4.5 检查变量的值	72
4.5.1 什么是表达式	72
4.5.2 监视表达式	72
4.5.3 计算和修改表达式	75
4.5.4 检查数据元素	76
4.5.5 检查寄存器的值	77
4.6 使用断点	77
4.6.1 设置断点	77
4.6.2 使用断点工作	78
4.6.3 用户定做断点和执行点	81
4.7 解决一般保护错误问题	81
4.8 使用 Event Log 窗口	81
4.9 调试动态连接库	82
第五章 Windows 95 编程简介	84
5.1 Windows 95 编程方式	84
5.1.1 桌面模型	84
5.1.2 鼠标	85
5.1.3 图标和位图	85
5.1.4 菜单、工具条、状态条和对话框	85
5.2 Windows 95 如何同用户应用程序通信	86
5.3 Win32 应用程序接口(API):	
Windows 95 应用程序接口(API)	86
5.4 窗口组件	87
5.5 Windows 95 应用程序基础	87
5.5.1 WinMain()	88
5.5.2 窗口函数	88
5.5.3 窗口类	88
5.5.4 消息循环	89
5.5.5 Windows 数据类型	89
5.6 Windows 95 程序框架	89

5.6.1 定义窗口类	92
5.6.2 创建窗口	94
5.6.3 消息循环(Message Loop)	95
5.7 窗口函数	97
5.8 使用定义文件	97
5.9 命名协议	98
第六章 处理消息	99
6.1 什么是消息	99
6.2 响应按键	99
6.3 设备场境	104
6.4 处理 WM_PAINT 消息	104
6.5 响应鼠标消息	108
6.6 产生 WM_PAINT 消息	113
6.7 产生计时器消息	117
第七章 开发菜单和键加速器	121
7.1 菜单机制	121
7.1.1 什么是菜单	121
7.1.2 菜单关键字和选项	122
7.2 随时创建菜单	126
7.2.1 创建菜单来改变图形的尺寸	126
7.2.2 用菜单改变背景颜色	131
7.2.3 用菜单决定系统信息	139
7.2.4 用菜单查看目录表列	147
7.3 总 结	156
第八章 数据输入对话框	157
8.1 对话框简介	160
8.2 Resource Workshop	161
8.2.1 为何要用 Dialog Editor	161
8.2.2 使用 BorlandDialog Editor	162
8.2.3 创建一个对话框	164
8.2.4 查看.DLG 文件	164
8.3 为各种需要创建对话框	166
8.3.1 创建一个简单的 About 对话框	166
8.3.2 用一个对话框改变图形形状	174
8.3.3 通过对话框输入文本	185
8.3.4 用对话框输入整型数	192
8.3.5 用对话框输入实型的数字	202
8.4 创建消息框	211

第九章 深入研究控件	217
9.1 使用复选框	217
9.2 管理复选框	223
9.2.1 开关复选框	223
9.2.2 初始化复选框	224
9.3 添加静态控件	228
9.4 增加单选按钮	229
9.5 使用滚动条控件	234
9.5.1 接收滚动条消息	235
9.5.2 设置滚动条范围	236
9.5.3 设置滚动条滑块框位置	236
9.5.4 滚动条程序实例	237
第十章 进程和线程多任务	243
10.1 创建特殊任务.....	243
10.2 创建多线程程序.....	252
10.2.1 创建线程.....	252
10.2.2 终止线程.....	253
10.2.3 简短的多线程例程.....	253
10.2.4 使用多线程.....	259
10.3 同 步.....	265
10.3.1 理解串行问题.....	266
10.3.2 Windows 95 同步对象	267
10.3.3 使用信号灯来同步线程.....	267
10.4 使用事件对象.....	275
10.5 需要进一步研究的内容.....	277
第十一章 ObjectWindows 综述	278
11.1 了解类的层次结构.....	278
11.1.1 使用类.....	278
11.1.2 继承成员.....	279
11.1.3 成员函数类型.....	280
11.2 对象的拓扑构造函数.....	281
11.2.1 窗口类.....	282
11.2.2 对话框类.....	282
11.2.3 控制类.....	283
11.2.4 图形类.....	284
11.2.5 打印类.....	285
11.2.6 模块和应用程序类.....	285
11.2.7 文档/视图类	285
11.2.8 其它杂类.....	285

11.3 应用程序对象	286
11.3.1 最低需求	286
11.3.2 初始化应用程序	288
11.3.3 应用程序消息处理	293
11.3.4 关闭应用程序	293
11.3.5 使用控制库	294
11.4 界面对象	295
11.4.1 界面对象有何用处	296
11.4.2 类属界面对象: TWindow	296
11.4.3 创建界面对象	296
11.4.4 删除界面对象	298
11.4.5 父与子界面元素	299
11.4.6 登录窗口类	303
11.5 事件处理	304
11.5.1 说明响应表	304
11.5.2 定义响应表	304
11.5.3 定义响应表入口	305
11.6 窗口对象	309
11.6.1 使用窗口对象	310
11.6.2 布局窗口	313
11.6.3 框架窗口	318
11.6.4 装饰框架窗口	320
11.6.5 MDI 窗口	321
11.7 菜单对象	324
11.7.1 创建菜单对象	324
11.7.2 调整菜单对象	325
11.7.3 查询菜单对象	326
11.7.4 使用系统菜单对象	326
11.7.5 使用弹出式菜单	326
11.7.6 在主窗口里添加菜单资源	327
11.8 对话框对象	327
11.8.1 使用对话框对象	327
11.8.2 使用一对话框作为主窗口	332
11.8.3 在对话框中操作控制命令	333
11.8.4 联系界面对象和控制	333
11.8.5 使用对话框	335
第十二章 循序渐进学习 ObjectWindows	344
12.1 开 始	344
12.2 步骤 1: 基本应用程序	344

12.3 步骤 2:处理窗口事件	346
12.3.1 增加一个窗口类	346
12.3.2 添加一个响应表	347
12.3.3 事件处理函数	348
12.3.4 被封装的 API 调用	348
12.3.5 重写 CanClose 函数	349
12.3.6 把 TMyWindow 当作主窗	349
12.3.7 哪里获得更多信息	350
12.4 步骤 3:在窗口内写	350
12.4.1 构造一个设备场境	350
12.4.2 设备场境下的显示	351
12.4.3 清除窗口	351
12.4.4 哪里查找更多信息	351
12.5 步骤 4:窗内作图	352
12.5.1 添加新事件	352
12.5.2 添加一个 TClientDC 指针	352
12.5.3 哪里查找更多信息	354
12.6 步骤 5:改变线的粗细	354
12.6.1 添加一个画刷	355
12.6.2 改变画刷尺寸	356
12.6.3 调用 SetPenSize	357
12.6.4 画刷的清除	358
12.6.5 哪里查找更多的信息	358
12.7 步骤 6:画窗和添加菜单	358
12.7.1 重绘窗口	359
12.7.2 菜单命令	363
12.8 步骤 7:使用通用对话框	365
12.8.1 改变 TMyWindow	365
12.8.2 改善 CanClose	366
12.8.3 CmFileSave 函数	367
12.8.4 CmFileOpen 函数	367
12.8.5 CmFileSaveAs 函数	368
12.8.6 打开和存图	369
12.8.7 CmAbout 函数	370
12.8.8 哪里查找更多信息	371
12.9 步骤 8:增加多条线	371
12.9.1 TLine 类	371
12.9.2 TLines 数组	372
12.9.3 插入和提取 TLine 对象	372

12.9.4 扩展 TMyWindow	373
12.9.5 何处获取更多信息	375
12.10 步骤 9: 改变画笔	375
12.10.1 TLine 类的改变	375
12.10.2 TMyWindow 类的改动	378
12.10.3 何处获取更多信息	379
12.11 步骤 10: 添加修饰	379
12.11.1 改变主窗口	380
12.11.2 创建状态条	380
12.11.3 创建控制条	381
12.11.4 在修饰框中插入对象	383
12.11.5 何处获取更多信息	383
12.12 步骤 11: 转向 Doc/View 模型	383
12.12.1 组织应用程序的源程序	384
12.12.2 Doc/View 模型	384
12.12.3 TDrawDocument 类	384
12.12.4 TDrawView 类	389
12.12.5 定义文档样本	392
12.12.6 支持应用程序中的 Doc/View	393
12.12.7 获取更多信息	397
12.13 步骤 12: 转向 MDI	397
12.13.1 在应用中支持 MDI	397
12.13.2 TDrawDocument 及 TDrawView 的改动	400
12.13.3 TDrawListView 类	409
12.13.4 获取更多信息	414
12.14 进一步学习的内容	414
第十三章 美化窗口设计	416
13.1 框架(Frame)和装饰	416
13.2 TDecorateFrame	416
13.3 状态条	418
13.4 控件条和构件	424
13.5 TLayoutWindow	434
13.6 命令允许	440
13.7 总 结	454
第十四章 活用高级控件	455
14.1 列表框	455
14.2 组合框	457
14.3 单选按钮和组框	467
14.4 滚动和滚动条	473

14.5	TScrollBar	474
14.6	TSlider 控件	476
14.7	TGauge 控件	476
14.8	总 结	482
第十五章	OLE2 简介	484
15.1	一些 OLE 术语	486
15.2	创建一个简单的 OLE 程序	486
15.3	什么是 Server ID 和为什么需要它们	488
15.4	建立用户的 OLE2 程序	489
15.5	调试 OLE2 程序	489
15.6	复合文件	491
15.6.1	复合文件的分段	493
15.6.2	存贮界面	495
15.6.3	IStorage 界面	496
15.6.4	IMalloc 界面	498
15.6.5	关于子存贮和流的函数	498
15.7	拖放界面	498
15.7.1	拖放源界面	499
15.7.2	拖放目标界面	500
15.7.3	拖放函数	501
15.8	原地激活和编辑界面	502
15.8.1	原地编辑界面	502
15.8.2	原地编辑结构	503
15.8.3	原地编辑函数	503
15.9	总 结	504
第十六章	创建 OLE2 应用程序	505
16.1	建立 OLE 服务器	505
16.2	建立 OLE 容器	518
16.3	总 结	525

第一章 Windows 95 概览

Microsoft Windows 操作系统为用户和开发人员提供了种类丰富的服务,它们充分利用了范围很广的有效硬件平台,这些平台包括从小的表格因子便携系统到多处理服务器。Windows 操作系统的可定标结构支持硬件范围日益扩大的相同用户接口、应用程序以及开发工具。同时,Windows 操作系统满足了用户各种各样的要求,从生产率应用程序到功能强大、安全的临界任务应用程序。

Windows 操作系统中的 Windows 95 操作系统的设计,着眼于主流系统。它是为基于 Intel 的台式机和便携机用户而设计的。对于主流系统,Windows 95 在保持所使用的系统资源量情况下,在一个较宽的应用程序范围内提供可靠的性能。Windows 95 是设计用于基于 Intel 的 PC,并且支持 Intel 80386DX、80486 以及用于主流台式机和便携 PC 的 Pentium 处理器的。

1.1 Windows 95 的最佳使用对象

对于用户来讲,下述情况下,Windows 95 操作系统将是最佳的选择:

(1) 在办公室环境下的多数需要人来执行的许多通用任务,譬如字处理、数据库查询或者数据表分析,使用生产率应用程序(如 Microsoft Office 程序组)。这些人也许使用用于其特殊商业的特定应用程序。对于已经安装了的个人计算机、外围设备和应用程序,Windows 95 将最大限度地利用在这些设施上的投资。

(2) 对于那些大部分工作时间都在办公室以外(例如,在生意现场,旅馆中或在野外),并且依赖个人计算机来帮助完成工作的用户来讲,Windows 95 将满足这些与其办公室同事有着相同应用程序和兼容性设备的移动计算机用户的需要,而同时只要求较低的硬件设置,包括内存容量、电池功率以及所用磁盘空间。

(3) 对于那些感到计算机既有挑战性又不太友好的家庭计算机用户,他们有时想要利用新的性能,譬如多媒体或在线信息服务。Windows 95 很容易让所有家庭成员来使用,并且允许做他们想做的任何事。在 Windows 95 操作系统中内建了丰富的多媒体功能,对游戏程序具有最高层次的兼容性,且具有与信息服务或其它在线服务的连通性。此外,即插即用技术能使用户在系统上自由地增减新部件,如打印机、调制解调器以及其它外围设备。

1.2 主要新特性

Windows 95 操作系统是微软公司的 Windows 3.1 操作系统的后继产品,它的出现是源于市场的需求。Windows 95 为满足市场需求,从特性方面所做不同改进从而来解决以前用户在使用 Windows 3.1 时遇到的问题。由于新特性的范围很宽,在此我们只列出在 Windows 3.1 不同邻域之上的改进之处。

1.2.1 易用性

(1) 对于终端用户

对于终端用户和类似 MIS 组织,在易用性方面 Windows 95 弥补了 Windows 3.1 中存在的不足。例如,缺乏经验的用户会发现重叠窗口和进行某项任务(如最大化、最小化窗口)太复杂,同时有经验的用户又希望提高工作效率。但是所作的改进不仅仅是解决这些问题,还包括对硬件、连通性和应用程序的改进。Windows 95 提供了以下解决方案:

- ① 一个崭新的用户接口(UI)。用户接口方面的一系列改进为所有用户大大增加了学习的简易性、使用能力以及学习效率,而不管用户的经验如何。初学者可以迅速入门,而有经验的用户可以充分利用自己的 PC。
- ② 即插即用(Plug and Play)。即插即用的技术使得用户在安装一个新的硬件设备之后,便可以马上使用。
- ③ 长文件名。取消了过去隐含的 8.3 文件名带来的束缚,长文件名只是 Windows 95 众多改进中的一个。

(2) 对于 MIS 组织

MIS 组织发现一个公司中的个人计算机数量的迅速增长越来越困难。此外,Windows 3.1 的引入带来了一系列的有关的安全性、可靠性以及管理方面的问题,这主要归因于它缺乏集成化的连通性和管理结构。而 Windows 95 系统中的内建网络、Registry、用户概貌、系统管理、用户层安全性以及网络备份代理等特性,解决了这些问题。

1.2.2 速度和功能

(1) 对于终端用户

与终端用户相关的主要领域就是提高使用 Windows 的效率和功能。用户总希望快速完成工作,希望能同时运行多个应用程序或计算机进程,而不是等待 PC 完成一个之后再启动另一个,也希望工作更加有效而不必牺牲系统的稳定性或工作性能。或许更重要的一点是用户不希望感到自己只利用了 PC 处理能力的一小部分。

Windows 95 的以下特性为用户带来更强的功能和更快的工作速度:

- ① 抢占先多任务。
- ② 可定标性能。
- ③ 支持 32 位应用程序。
- ④ 增强的坚固性。
- ⑤ 可移动计算。
- ⑥ 快速打印。
- ⑦ 高性能多媒体。
- ⑧ 为基于 MS-DOS 的应用程序提供更多的内存。

(2) 对于 MIS 组织

由于 MS-DOS 和 Windows 并非为网络环境所设计,因此在这些环境下会产生一连串问

题。这些问题包含没有足够的实模式内存用于运行基于 MS-DOS 的特定任务应用程序,在 Windows 3.1 下不稳定等。

Windows 95 的以下特性解决了这些问题,它们使得 MIS 组织的工作变得更加容易:

- ① 32 位 NetWare 连通性。
- ② 多网络支持。
- ③ 通用信息客户。

1.2.3 兼容性

(1) 对于终端用户

假如一个操作系统的升级需要新的软件、更多内存或新的硬件,那么升级的费用将比购买新机器的价格还要高。Windows 95 的最大设计目的之一就是让每一个用户现在就能用到最新的 Windows 版本。它的以下特性就是基于这种设计思想:

- ① 与现有的基于 MS-DOS 和基于 Windows 的应用程序兼容。
- ② 相同或更好的运行性能。
- ③ 对于出现的硬件设备具有后向兼容性。

(2) 对于 MIS 组织

从花费上考虑,MIS 组织要升级为新的操作系统还是存在问题的。然而,保持与现有硬件和软件的兼容性将更为重要,这不仅仅是由于升级规模更大,而且还由于在组织的不同群体中可能使用的操作系统并不相同。由于必须支持多个操作系统,使得 MIS 组织面临更大的问题。

Windows 95 通过以下特性来满足这类需求:

- ① 网络兼容性。
- ② 一种自调节指南 Windows 3.1 帮助。
- ③ 包含 Windows 3.1 的 Program Manager 和 File Manager。

1.3 Windwos 95 的用户接口

当用户首次启动 Windows 95 操作系统时将会立即发现它已不再是老式的在 MS-DOS 之上运行的 Windows 了。那些只有对一小部分计算机用户有意义的字符模式引导信息已经不存在了。取而代之的是以图形方式进入新的 Windows 95 用户接口(UI)桌面。

用户接口比操作系统的其它部分更能决定用户的整体经验。用户接口越简单、功能越强、越友好,则用户操作计算机时感觉就会越好,工作效率也就越高。优秀的用户接口能帮助用户顺利进行,是因为它使得对于包括初学者和熟练用户在内的所有人来说,计算机更简单更自然。

Windows 95 中用户接口的最主要设计目标是使得 PC 对所有人来说者都易于使用。